Stage 1

For this stage of the application, you will provide the XHTML structure and CSS presentation styles for all of the pages required by the application. Since no java coding is being used at this point, mock data will be used for all page content (i.e., make it all up and hard code it into the page). Where pages link to other pages, they will link to static content.

The project is divided up into **Group A** and a **Group B** responsibilities. You are responsible for producing <u>all</u> of the work for the group that you are in. Your partner is responsible for the work in the opposite group (a team is made up of a Group A and Group B member). So, while you and your partner make one submission which has both parts combined, your work is yours and your partner's work is your partners. You and your partner may use a common CSS file for setting site-wide presentation styles.

This stage is worth **3%** of the 15% project mark. Marks will be divided 65% for document structure and 35% for style. Submit both parts (Group A and Group B) into a single Eclipse Dynamic Web Project. Zip the entire project folder. Make sure that each of your pages vaidates with the [The W3C Markup Validation Service](#) and that each of your CSS files validates with the [The W3C CSS Validation Service](#). Submit the project to the submission drop box by 11:59PM on **Thursday October 22, 2015**. Only submit one file per group, but be sure to include the your name *within* the XHTML files that you author as XHTML <!-- comment --> tags. Note that the XML declaration (<?xml version="1.0" encoding="UTF-8" ?>) must be the first line in the file. Include the comment after that.

To help keep the project uniform, the names of the JavaServer Page files are given below along with a description of what that page should contain.

Group A

Members of this group are responsible for the following pages:

librarylogin.jspx
This page displays an HTML form with one text input for the username and one password input for the password. The password must use a password-type input that does not show the typed characters. The form also contains a submit button that the library account owner will click to login. The action page for the HTML form is **librarylogin.jspx** (i.e. the page round-trips). This form should use the POST method.

The page should contain a link to the **newlibraryaccount.jspx** page inviting a non library account owner to create an account.

For this stage, the page should also display both the welcome message inviting the library account owner to log into the Library Material Reservation Application as well as the error message displayed if the library account owner's login attempt has failed.

This page should not contain the common menu.

newlibraryaccount.jspx
This page contains an HTML form prompting for the non library account owner's first name, last name,

username, password (a password input that does not show the typed characters), and a confirm password (also a password input). A submit button is provided to submit the non library account owner's request. The first name and last name inputs should restrict the entered text to a maximum of **50** characters. The username, password, and confirm password inputs should restrict the entered text to a maximum of **15** characters. The action for the form is **newlibraryaccount.jspx** (i.e., the page round-trips).

For this stage, the page should also display the following five error messages (the text does not have to be exactly the same):

- First name is not valid.
- Last name is not valid.
- Username is not valid.
- Password is not valid.
- The password and the confirm password do not match.

The error messages should be highlighted in some way to indicate that they are error messages.

These are the possible validation errors that the page may display. This form should use the POST method.

The page includes a link to take the user back to the context root of the application (**href="."**).

This page should not contain the common menu.

showmaterialreservations.jspx
For this stage, the page should include the message stating that the library account owner currently has no library material reservations (as later this will be the message displayed if there are no library material reservations). The page should also show the title, the type of the material, and the reservation expriation date of all of the library material reservations for the library account owner. Each library material reservation displayed should also contain a separate HTML form with a submit button for cancelling the reservation. So if there are 5 library material reservations listed, there are 5 separate forms each with its own submit button. To identify the library material reservation being cancelled, the material id is specified in a hidden input within each form. For this stage, the title, the material type (Book, Audio, Video, etc.), the reservation expiration date, and the material id should be hardcoded into the page. You should hardcode at least two rows The action for each form is **cancelreservationconfirmation.jspx**. Each form should use the POST method.

This page should also contain a common menu. The menu will be provided by the file **limaramenu.jspx**. This page must simply *include* the menu page.

badrequest.jspx
This page displays an error message stating that the request specified is not valid. The page includes a link to take the library account owner back to the context root of the application (**href="."**).

This page should not contain the common menu.

Group B

Members of this group are responsible for the following pages:

limaramenu.jspx
The menu is a page fragment that is to be included in other pages (see the menu in the SIMA project for an example of what the page fragment should look like). The menu should show the library account owner's full name. The menu should also show the number of library materials currently reserved for the library account owner (show **0** if there are none). For this stage, hardcode the library account owner's name and the number of library materials currently reserved for the library account owner. The menu should include a link to reserve library materials (**reservematerials.jspx**), a link to view the library account owner's existing library material reservations (**showmaterialreservations.jspx**), and a link to log out (**logout.jspx**). Note that **logout.jspx** is a fictitious page.

reservematerials.jspx
This page contains an HTML form prompting for the title (or partial title) and the type of the desired library materials (i.e., the search criteria). A button is provided to submit a request to check what's available. Use a text box for the (partial) title. The library material type parameter should be specified by a drop down list (<select> element) allowing for the options **All**, **Book**, **Audio**, and **Video** (plus any other ones that you care to add). **All** should be first in the list. The action for the form is **reservematerials.jspx** (i.e., the page round-trips).

For this stage, the page should also display the error message (the text does not have to be exactly the same):

* The title cannot be empty.

The error message should be highlighted in some way to indicate that it is an error message.

This is the possible validation error that the page may display. This form should use the POST method.

Below the library material reservation details is the list of available library materials that meet the search criteria. This list only appears if the library account owner has submitted an available library materials check request. The tabular list shows the title and type of all available library materials matching the search criteria. Each library material in the list contains a separate HTML form with a submit button for reserving the library material for the library account owner. So if there are 6 available library materials in the list, there are 6 separate forms each with its own submit button. To identify the available library material, the material identification number (ISBN/UPC/ISAN) is specified in a hidden input within each form. The form must also contain a hidden input for the (partial) title and library material type. The action for the form is **reservematerials.jspx** (i.e., the page round-trips). This form should use the POST method. For this stage, always show the available library materials list. The material identification number (ISBN/UPC/ISAN), the (partial) title, and library material type parameters should be hardcoded into the page. You should hardcode at least two available library materials. Be sure to use the *same parameter names* for the (partial) title and library material type as you used in the search portion above.

Above the list of available library materials there should be a heading that indicates the search criteria for the available library materials ((partial) title and library material type).

This page should also contain a common menu. The menu will be provided by the file **limaramenu.jspx**. This page must simply *include* the menu page.

cancelreservationconfirmation.jspx
This is a response page that is displayed after the library account owner has submitted a library material reservation cancellation request. The page should display the title and type of the library material and ask the library account owner if they are sure that they want to cancel the library material reservation. An

HTML form is used to get the library account owner's confirmation response. The form has two submit buttons to allow the library account owner to confirm the cancellation or abort the cancellation. To identify the library material reservation to be cancelled, the material id is specified in a hidden input within the form. For this stage, the title, library material type, and material id should be hardcoded into the page. The action for the form is **cancelreservationconfirmation.jspx** (i.e., the page round-trips). This form should use the POST method.

This page should not contain the common menu.

Back to Overview Stage 2